mecloglog postestimation — Postestimation tools for mecloglog

Description

The following postestimation command is of special interest after `mecloglog`:

Command	Description
`estat group`	summarize the composition of the nested groups

The following standard postestimation commands are also available:

Command	Description
`contrast`	contrasts and ANOVA-style joint tests of estimates
`estat ic`	Akaike's and Schwarz's Bayesian information criteria (AIC and BIC)
`estat summarize`	summary statistics for the estimation sample
`estat vce`	variance–covariance matrix of the estimators (VCE)
`estimates`	cataloging estimation results
`lincom`	point estimates, standard errors, testing, and inference for linear combinations of coefficients
`lrtest`	likelihood-ratio test
`margins`	marginal means, predictive margins, marginal effects, and average marginal effects
`marginsplot`	graph the results from margins (profile plots, interaction plots, etc.)
`nlcom`	point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients
`predict`	predictions, residuals, influence statistics, and other diagnostic measures
`predictnl`	point estimates, standard errors, testing, and inference for generalized predictions
`pwcompare`	pairwise comparisons of estimates
`test`	Wald tests of simple and composite linear hypotheses
`testnl`	Wald tests of nonlinear hypotheses

Special-interest postestimation commands

`estat group` reports the number of groups and minimum, average, and maximum group sizes for each level of the model. Model levels are identified by the corresponding group variable in the data. Because groups are treated as nested, the information in this summary may differ from what you would get if you used the `tabulate` command on each group variable individually.

Syntax for predict

Syntax for obtaining predictions of random effects and their standard errors

```
predict [type] newvarsspec [if] [in], { remeans | remodes } [reses(newvarsspec)]
```

Syntax for obtaining other predictions

```
predict [type] newvarsspec [if] [in] [, statistic options]
```

newvarsspec is *stub** or *newvarlist*.

statistic	Description
Main	
`mu`	predicted mean; the default
`fitted`	fitted linear predictor
`xb`	linear predictor for the fixed portion of the model only
`stdp`	standard error of the fixed-portion linear prediction
`pearson`	Pearson residuals
`deviance`	deviance residuals
`anscombe`	Anscombe residuals

These statistics are available both in and out of sample; type `predict ... if e(sample) ...` if wanted only for the estimation sample.

options	Description
Main	
`means`	compute *statistic* using empirical Bayes means; the default
`modes`	compute *statistic* using empirical Bayes modes
`nooffset`	ignore the offset variable in calculating predictions; relevant only if you specified `offset()` when you fit the model
`fixedonly`	prediction for the fixed portion of the model only
Integration	
`intpoints(#)`	use # quadrature points to compute empirical Bayes means
`iterate(#)`	set maximum number of iterations in computing statistics involving empirical Bayes estimators
`tolerance(#)`	set convergence tolerance for computing statistics involving empirical Bayes estimators

Menu for predict

Statistics > Postestimation > Predictions, residuals, etc.

Options for predict

Main

remeans, remodes, reses(); see [ME] **meglm postestimation**.

mu, the default, calculates the predicted mean (the probability of a positive outcome), that is, the inverse link function applied to the linear prediction. By default, this is based on a linear predictor that includes both the fixed effects and the random effects, and the predicted mean is conditional on the values of the random effects. Use the fixedonly option if you want predictions that include only the fixed portion of the model, that is, if you want random effects set to 0.

fitted, xb, stdp, pearson, deviance, anscombe, means, modes, nooffset, fixedonly; see [ME] **meglm postestimation**.

By default or if the means option is specified, statistics mu, fitted, xb, stdp, pearson, deviance, and anscombe are based on the posterior mean estimates of random effects. If the modes option is specified, these statistics are based on the posterior mode estimates of random effects.

Integration

intpoints(), iterate(), tolerance(); see [ME] **meglm postestimation**.

Syntax for estat group

```
estat group
```

Menu for estat

Statistics > Postestimation > Reports and statistics

Remarks and examples

stata.com

Various predictions, statistics, and diagnostic measures are available after fitting a mixed-effects complementary log-log model with mecloglog. Here we show a short example of predicted probabilities and predicted random effects; refer to [ME] **meglm postestimation** for additional examples.

▷ Example 1

In example 2 of [ME] **mecloglog**, we analyzed the cognitive ability (dtlm) of patients with schizophrenia compared with their relatives and control subjects. We used a three-level complementary log-log model with random effects at the family and subject levels. Cognitive ability was measured as the successful completion of the "Tower of London", a computerized task, measured at three levels of difficulty.

```
. use http://www.stata-press.com/data/r13/towerlondon
(Tower of London data)
. mecloglog dtlm difficulty i.group || family: || subject:
Fitting fixed-effects model:
  (output omitted )
Mixed-effects cloglog regression                Number of obs      =       677

-------------------------------------------------------------
                |   No. of       Observations per Group
 Group Variable |   Groups    Minimum    Average    Maximum
----------------+--------------------------------------------
         family |      118          2        5.7         27
        subject |      226          2        3.0          3
-------------------------------------------------------------

Integration method: mvaghermite                 Integration points =         7

                                                Wald chi2(3)       =     83.32
Log likelihood = -305.26516                     Prob > chi2        =    0.0000
------------------------------------------------------------------------------
        dtlm |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
  difficulty |  -1.342844   .1501508    -8.94   0.000    -1.637135   -1.048554
             |
       group |
          2  |  -.1331007    .269389    -0.49   0.621    -.6610935    .3948922
          3  |  -.7714314   .3097099    -2.49   0.013    -1.378452    -.164411
             |
       _cons |    -1.6718   .2290325    -7.30   0.000    -2.120695   -1.222905
-------------+----------------------------------------------------------------
family       |
   var(_cons)|   .2353453   .2924064                      .0206122    2.687117
-------------+----------------------------------------------------------------
family>      |
subject      |
   var(_cons)|   .7737687   .4260653                      .2629714    2.276742
------------------------------------------------------------------------------
LR test vs. cloglog regression:      chi2(2) =    16.61   Prob > chi2 = 0.0002
Note: LR test is conservative and provided only for reference.
```

We obtain predicted probabilities based on the contribution of both fixed effects and random effects by typing

```
. predict pr
(predictions based on fixed effects and posterior means of random effects)
(option mu assumed)
(using 7 quadrature points)
```

As the note says, the predicted values are based on the posterior means of random effects. You can use the modes option to obtain predictions based on the posterior modes of random effects.

We obtain predictions of the posterior means themselves by typing

```
. predict re*, remeans
(calculating posterior means of random effects)
(using 7 quadrature points)
```

Because we have one random effect at the family level and another random effect at the subject level, Stata saved the predicted posterior means in the variables re1 and re2, respectively. If you are not sure which prediction corresponds to which level, you can use the describe command to show the variable labels.

Here we list the data for family 16:

```
. list family subject dtlm pr re1 re2 if family==16, sepby(subject)

      +-------------------------------------------------------------+
      | family   subject   dtlm         pr         re1         re2 |
      |-------------------------------------------------------------|
 208. |     16         5      1    .486453    .4184933    .2760492 |
 209. |     16         5      0   .1597047    .4184933    .2760492 |
 210. |     16         5      0   .0444156    .4184933    .2760492 |
      |-------------------------------------------------------------|
 211. |     16        34      1   .9659582    .4184933    1.261488 |
 212. |     16        34      1   .5862808    .4184933    1.261488 |
 213. |     16        34      1    .205816    .4184933    1.261488 |
      |-------------------------------------------------------------|
 214. |     16        35      0   .5571261    .4184933   -.1616545 |
 215. |     16        35      1   .1915688    .4184933   -.1616545 |
 216. |     16        35      0   .0540124    .4184933   -.1616545 |
      +-------------------------------------------------------------+
```

We can see that the predicted random effects (re1) at the family level are the same for all members of the family. Similarly, the predicted random effects (re2) at the individual level are constant within each individual. Based on a cutoff of 0.5, the predicted probabilities (pr) for this family do not match the observed outcomes (dtlm) as well as the predicted probabilities from the logistic example; see example 1 in [ME] **melogit postestimation**.

◁

Methods and formulas

Methods and formulas for predicting random effects and other statistics are given in *Methods and formulas* of [ME] **meglm postestimation**.

Also see

[ME] **mecloglog** — Multilevel mixed-effects complementary log-log regression

[ME] **meglm postestimation** — Postestimation tools for meglm

[U] **20 Estimation and postestimation commands**